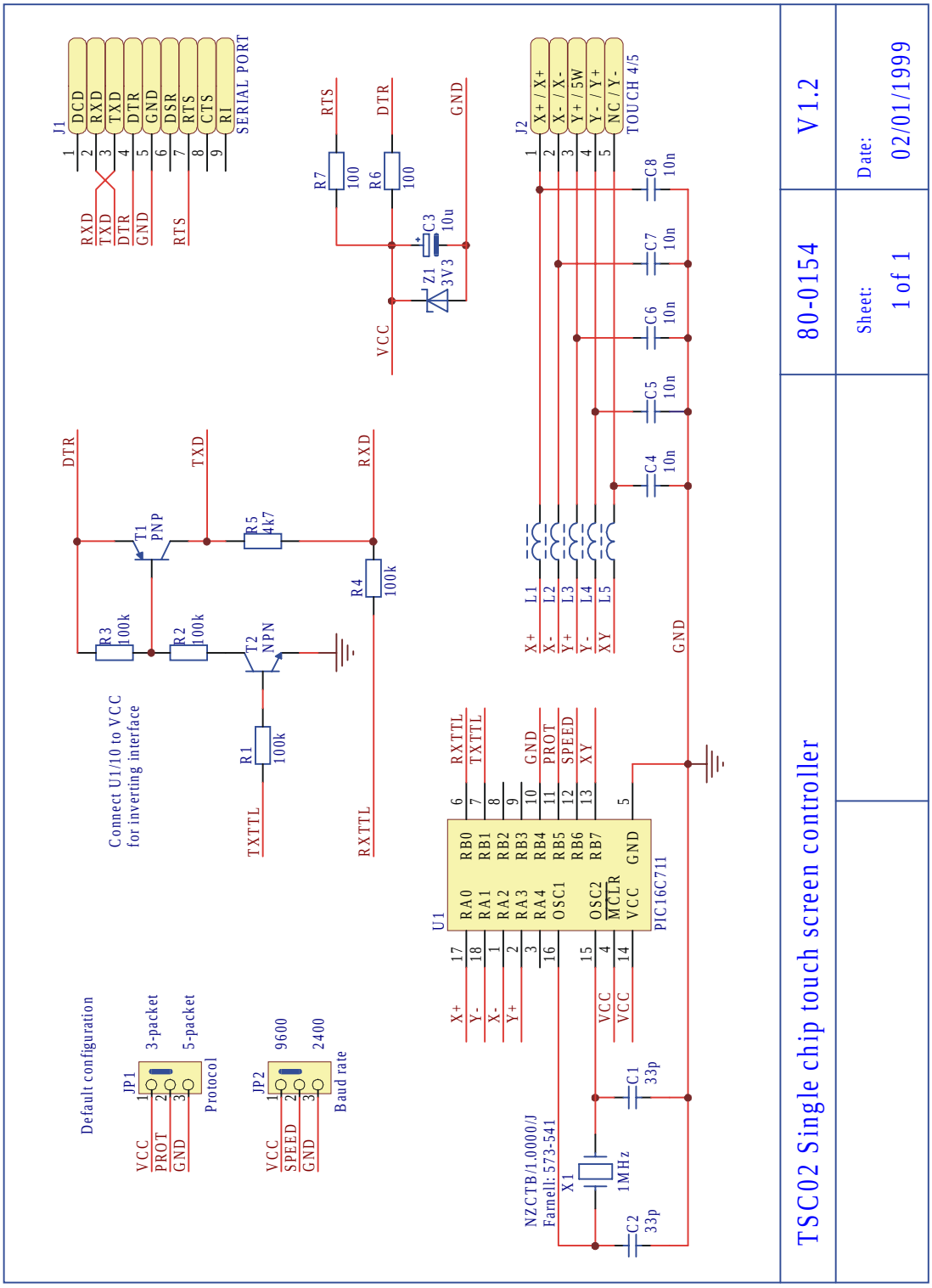
# TSC02
# Touch Screen
# Controller Card

# Contents

**Contents** ..... **3**
**1. Introduction** ..... **4**
**2. Hardware Description** ..... **5**
2.1. Serial Connector ..... 5
2.2. Serial Interface ..... 5
2.3. Microcontroller ..... 6
2.4. Power Supply ..... 6
2.5. Touch Screen Connector ..... 7
2.6. Default Configuration Jumpers ..... 7
**3. Parameters** ..... **8**
3.1. Filtering ..... 8
3.2. Hi/Lo filtering ..... 8
3.3. Number of wires ..... 8
3.4. Min/Max ..... 8
3.5. Touch mode ..... 8
3.6. Baud rate ..... 9
3.7. Protocol ..... 9
**4. Communication** ..... **10**
4.1. Command word 00 ..... 11
4.2. Command word 01 ..... 11
4.3. Read back ..... 11
**5. Test programs** ..... **12**
5.1. TCOM ..... 12
5.2. TDRAW ..... 12
**6. Troubleshooting** ..... **13**

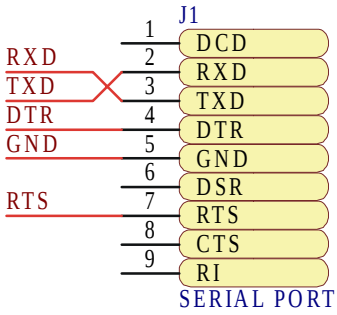
# 1. Introduction

TSC02 is a high speed single chip touch screen controller. It is usually integrated into the LCD controller card or to the LCD itself, but available as a stand-alone evaluation card too. This manual explains the connection and working of the stand-alone card.



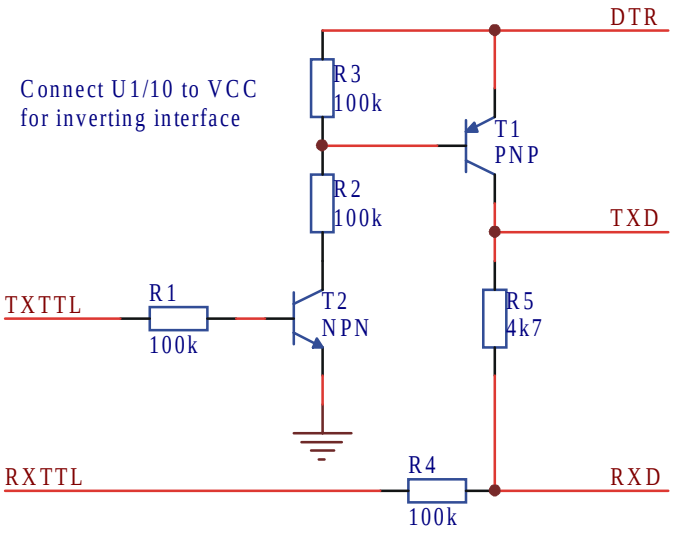
# 2. Hardware Description

## 2.1. Serial Connector



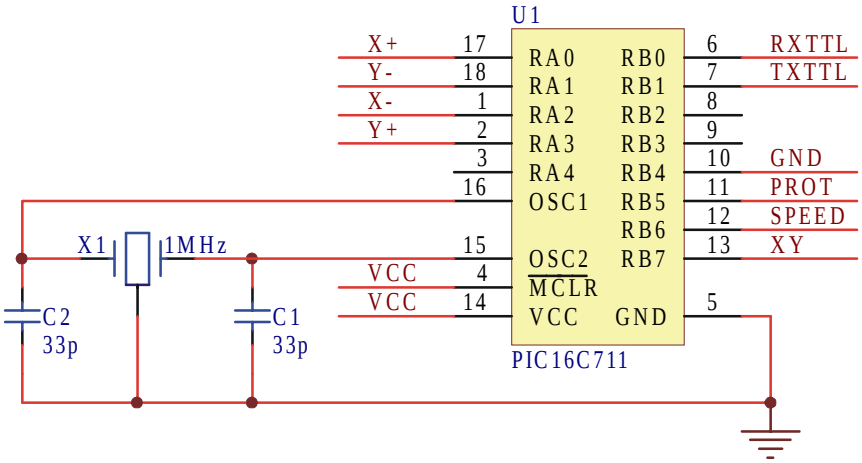

TSC02 is connected to the RS232 interface by a standard 9-pin D-type connector. Only the RXD and TXD lines are used for communication. DTR is used for power supply generation.

## 2.2. Serial Interface



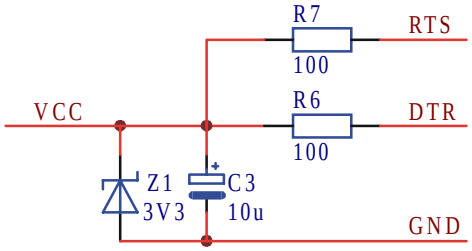

The serial interface is a simple level shifter circuit, similar to those used in mice. The microcontroller can use either normal or inverted signals. Connect U1/10 to VCC, if you are using a true inverting RS232 driver IC.

## 2.3. Microcontroller



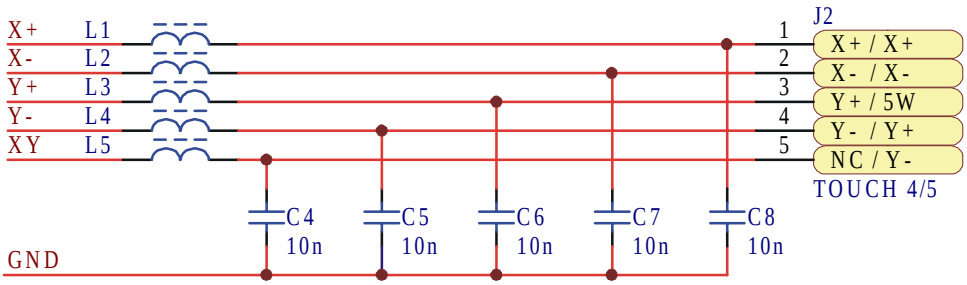

The heart of the TSC02 is the microcontroller. It handles the touch screen and communicates with the host computer

## 2.4. Power Supply

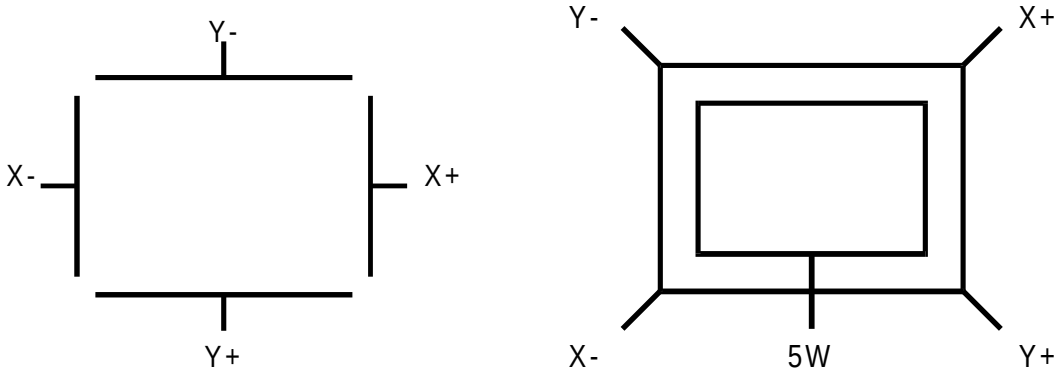



Power supply is derived from the DTR and RTS line. For normal operation DTR and RTS must be set to high. The current required is 20mA. VCC is 3-5.5V.

## 2.5. Touch Screen Connector

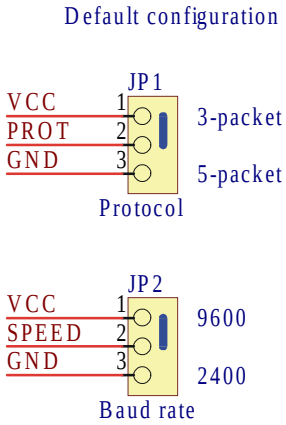



The touch screen is connected to a pin header. TSC02 supports 4 and 5 wire touch screens. The next picture shows the connection method.




Front view

## 2.6. Default Configuration Jumpers




The power-up default configuration is determined by JP1 nad JP2 jumpers. If the TSC02 controller is integrated into the LCD these jumpers are usually hard wired.

# 3. Parameters

The operation of the TSC02 is controlled by a set of parameter.

## 3.1. Filtering

The filtering option selects the number of samples the controller take at each point. The average of these samples determine the touch position. Higher value means lower speed, but better accuracy. Default value: 32.

## 3.2. Hi/Lo filtering

Hi/Lo filtering uses lower filtering value at high velocities, and higher filtering value at low velocities. It improves the overall performance of the controller. Default: ON.

## 3.3. Number of wires

TSC02 currently supports only 4 wire touch screen interface. If TSC02 supports 5, 7 and 8 wire interface in the future then automatic recognition will be used. The command field that changes the number of wires is ignored.

## 3.4. Min/Max

Min/Max controls the sensitivity of TSC02. In high noise environment TSC02 works more reliably, though less accurately, if set to HIGH. Default: HIGH.

## 3.5. Touch mode

*Touch mode*

A continuous stream of **pen down** codes is sent at touch down.
One **pen up** code at touch up.
This is the default mode.

*Untouch up mode:*

No code is sent at touch down
One **pen down** and immediately one **pen up** code is sent at touch up.

*Untouch down mode*

One **pen down** and immediately one **pen up** code is sent at touch down.
No code is sent at touch up.

## 3.6. Baud rate

TSC02 supports two baud rates: 2400 baud and 9600 baud, however at 2400 baud it works too slowly, therefore it is not recommended. Be careful when switching baud rate, because the command is acknowledged at the old baud rate, and only after that will the new baud rate be used. The default value is determined by the connection of pin 12 of the TSC02 controller. If pin 12 is connected to VCC, then the default baud rate is 9600. If pin 12 is connected to GND, then the default baud rate is 2400.

## 3.7. Protocol

Most programs uses the 3-packet protocol nowadays. The default value is determined by the connection of pin 11 of the TSC02 controller. If pin 11 is connected to VCC, then the default protocol is 3-packet. If pin 11 is connected to GND, then the default protocol is 5-packet.

*3-packet protocol (also called new mode).*

**pen down** code

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Sync | 1 | 1 | X9 | X8 | X7 | Y9 | Y8 | Y7 | Packet 1 |
| Data | 0 | X6 | X5 | X4 | X3 | X2 | X1 | X0 | Packet 2 |
| Data | 0 | Y6 | Y5 | Y4 | Y3 | Y2 | Y1 | Y0 | Packet 3 |

**pen up** code

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Sync | 1 | 0 | X9 | X8 | X7 | Y9 | Y8 | Y7 | Packet 1 |
| Data | 0 | X6 | X5 | X4 | X3 | X2 | X1 | X0 | Packet 2 |
| Data | 0 | Y6 | Y5 | Y4 | Y3 | Y2 | Y1 | Y0 | Packet 3 |

*5-packet protocol (also called old mode)*

**pen down** code

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Sync | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Packet 1 |
| Data | 0 | 0 | 0 | 0 | 0 | 0 | X1 | X0 | Packet 2 |
| Data | X9 | X8 | X7 | X6 | X5 | X4 | X3 | X2 | Packet 3 |
| Data | 0 | 0 | 0 | 0 | 0 | 0 | Y1 | Y0 | Packet 4 |
| Data | Y9 | Y8 | Y7 | Y6 | Y5 | Y4 | Y3 | Y2 | Packet 5 |

**pen up** code

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Sync | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Packet 1 |
| Data | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | Packet 2 |
| Data | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | Packet 3 |

# 4. Communication

The TSC02 communicates at 9600 baud, 8 data bits, 1 stop bit, no parity format over the RS232 serial line. Parameters can be set the following way:

1. Send SOH code (01H).
2. Skip the echoed SOH code (01H).
3. Wait for ACK code (06H).
4. If no ACK is received for 100 ms then the TSC02 is not connected.
5. Within 100 ms send the command code.
6. Skip the echoed command code.
7. Wait for ACK code (06H).
8. If no ACK is received for 100 ms then the TSC02 is not connected.
9. Within 100 ms send CR code (0DH).
10. Skip the echoed CR code (0DH).
11. Wait for ACK code (06H).
12. If no ACK is received for 100 ms then the TSC02 is not connected.

Please note that the RS232 transmit and receive lines are connected in a way that you receive every sent byte in the same time.

## 4.1. Command word 00

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Filtering | 0 | 0 | | | | | 0 | 0 | 4 |
| Filtering | 0 | 0 | | | | | 0 | 1 | 8 |
| Filtering | 0 | 0 | | | | | 1 | 0 | 16 |
| Filtering | 0 | 0 | | | | | 1 | 1 | 32 |
| Hi/Lo filter | 0 | 0 | | | | 0 | | | Off |
| Hi/Lo filter | 0 | 0 | | | | 1 | | | On |
| # Wires | 0 | 0 | | 0 | 0 | | | | 7 |
| # Wires | 0 | 0 | | 0 | 1 | | | | 5 |
| # Wires | 0 | 0 | | 1 | 0 | | | | 8 |
| # Wires | 0 | 0 | | 1 | 1 | | | | 4 |
| Min/Max | 0 | 0 | 0 | | | | | | HIGH |
| Min/Max | 0 | 0 | 1 | | | | | | LOW |

## 4.2. Command word 01

| Bit # | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Description |
|---|---|---|---|---|---|---|---|---|---|
| Mode | 0 | 1 | | | | | 0 | 0 | Reserved |
| Mode | 0 | 1 | | | | | 0 | 1 | Untouch up |
| Mode | 0 | 1 | | | | | 1 | 0 | Untouch down |
| Mode | 0 | 1 | | | | | 1 | 1 | Touch |
| Baud rate | 0 | 1 | | | 0 | 0 | | | Reserved |
| Baud rate | 0 | 1 | | | 0 | 1 | | | 2400 |
| Baud rate | 0 | 1 | | | 1 | 0 | | | Reserved |
| Baud rate | 0 | 1 | | | 1 | 1 | | | 9600 |
| Protocol | 0 | 1 | | 0 | | | | | 5-packet |
| Protocol | 0 | 1 | | 1 | | | | | 3-packet |
| Reserved | 0 | 1 | 0 | | | | | | Reserved |
| Reserved | 0 | 1 | 1 | | | | | | Reserved |

## 4.3. Read back

The 0F0H command requests parameter readback. After acknowledging the command TSC02 returns 3 bytes:

1. Command word 0 (# Wires are set to the autodetected value)
2. Command word 1
3. Version number (V1.3 is returned as 12H)

Additional bytes may be returned by future versions.

# 5. Test programs

Test programs are written for DOS. Source code is also available.

## 5.1. TCOM

TCOM simply displays data received from the touch overlay in hex and text format. It is also capable of sending data to touch overlay. Any byte entered on the keyboard are sent to the touch controller as command code enclosed in SOH and CR. The utility accesses the touch overlay controller directly, without any drivers or filtering.

Usage:

TCOM /A:address /I:irq

or:

TCOM /n

where address is the address of the serial line the touch overlay controller is connected to, and irq is the interrupt request line number. Standard COMn serial port can also be selected by /n.

## 5.2. TDRAW

TDRAW interprets the coordinates and shows them graphically. This utility accesses the touch overlay controller directly, without any drivers or filtering. This program requires a standard VGA card.

Usage:

TDRAW /A:address /I:irq

or:

TDRAW /n

where address is the address of the serial line the touch overlay controller is connected to, and irq is the interrupt request line number. Standard COMn serial port can also be selected by /n.

There is no way of calibrating the program. It is normal that the area of the touch screen does not fill the entire screen. TDRAW works in 3-packet mode only.

# 6. Troubleshooting

If the touch screen controller does not work, please follow the checklist:

1. Restart the computer in DOS only mode.
2. Set the baud rate. (MODE COM1:9600,n,8,1)
3. Start TCOM. Use port and interrupt settings you think good.
4. Read back parameters by sending F0.
   The touch screen should return something like
   01 06 F0 06 0D 06 3F 5F 12
5. If you receive nothing, then the touch screen controller is not connected to the serial line you try to use, or uses a different IRQ.
6. If you receive 01 only, then the touch screen controller is connected, but uses different speed.
7. Interpret the returned command words.
   3F: HIGH Min/Max filter, 4 wire interface, Hi/Lo filter ON, filtering 32.
   5F: 3-packet protocol, 9600 baud, touch mode.
8. Touch the screen, and check the returned bytes. If nothing happens then the touch screen is not connected properly.
9. Set touch mode, 3-packet protocol, and exit TCOM.
10. Start TDRAW, and draw something nice. It is normal that the used area does not fill the entire screen. The screen may display mirrored image, depending the connection method. But if you draw a line on the touch screen then you should see a dotted line on the display screen.

If the touch screen works with TCOM and TDRAW, but not in the environment you try to use it then the drivers are not installed properly. Please refer to the driver documentation.